Necesarry libraries

```
library(readr)
library(dplyr)
```

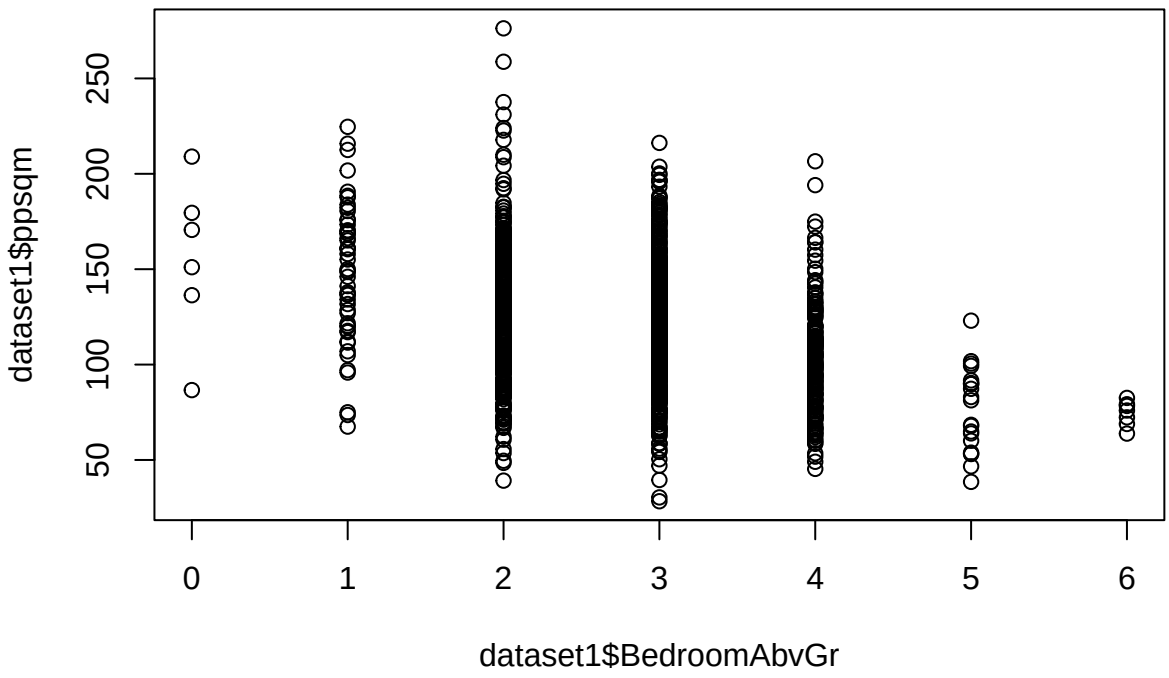
```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

Loading dataset

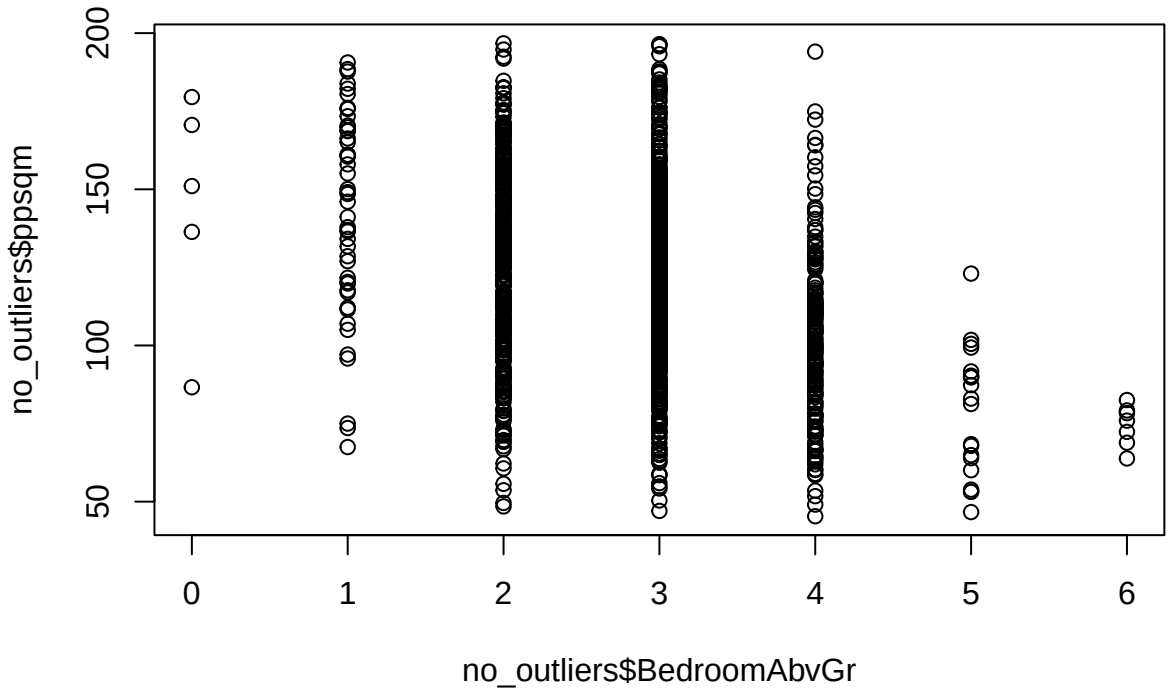
```
dataset <- read_csv("preprocessed_data.csv")
```

```
## Rows: 1460 Columns: 81
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (43): MSZoning, Street, Alley, LotShape, LandContour, Utilities, LotConf...
## dbl (38): Id, MSSubClass, LotFrontage, LotArea, OverallQual, OverallCond, Ye...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
dataset1 = subset(dataset, BedroomAbvGr != 8)

dataset1$ppsqm = dataset1$SalePrice / dataset1$GrLivArea

plot(dataset1$BedroomAbvGr,dataset1$ppsqm)
```



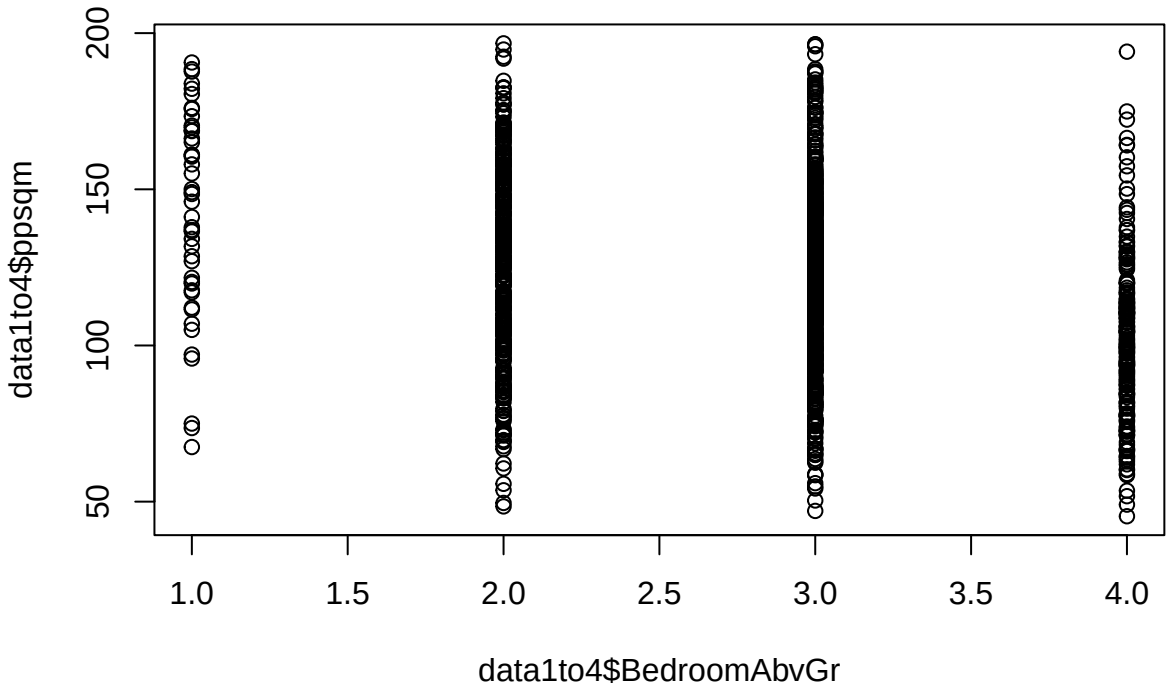

```
quartiles = quantile(dataset1$ppsqm, probs = c(.25, .75), na.rm=FALSE)
IQRppsqm = IQR(dataset1$ppsqm)

lower <- quartiles[1] - 1.5*IQRppsqm
upper <- quartiles[2] + 1.5*IQRppsqm

no_outliers = subset(dataset1, dataset1$ppsqm > lower & dataset1$ppsqm < upper)

plot(no_outliers$BedroomAbvGr, no_outliers$ppsqm)
```



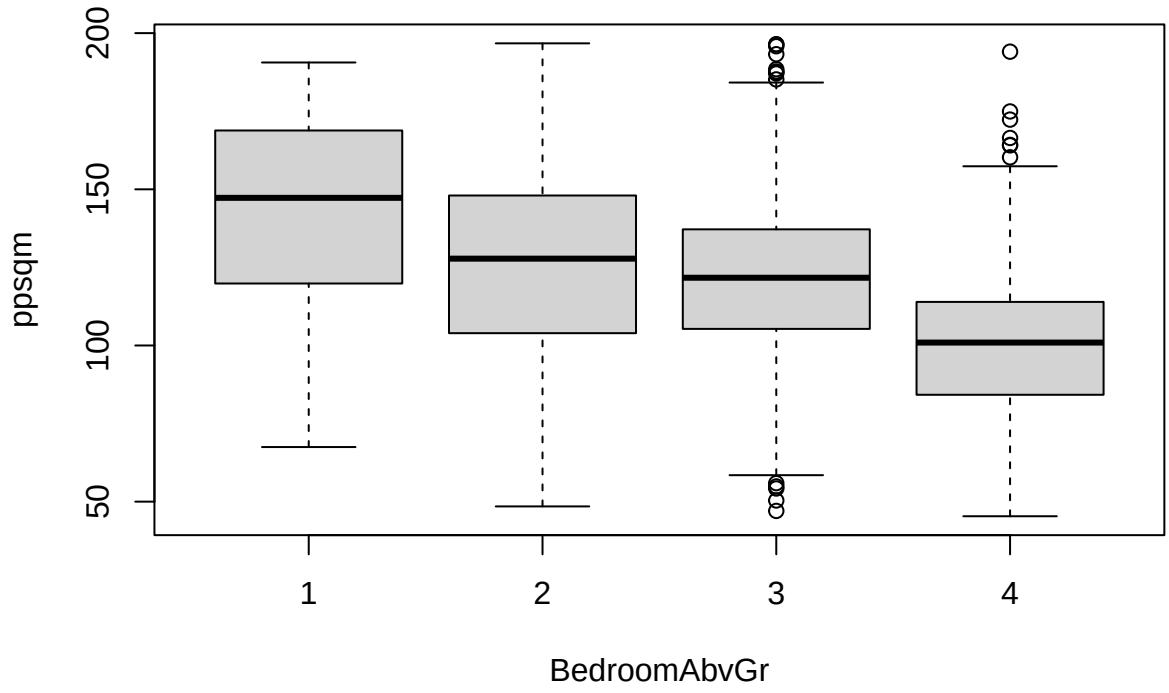

```
dataset2 <- dataset1[names(dataset1) %in% c('ppsqm', 'BedroomAbvGr')]

no_outliers %>%
  group_by(BedroomAbvGr) %>%
  count() -> dataset3

dataset3
```

```
## # A tibble: 7 x 2
## # Groups:   BedroomAbvGr [7]
##   BedroomAbvGr     n
##          <dbl> <int>
## 1            0     5
## 2            1    46
## 3            2   347
## 4            3   796
## 5            4   212
## 6            5    20
## 7            6     7
```

```
data1to4 = subset(no_outliers, no_outliers$BedroomAbvGr > 0 & no_outliers$BedroomAbvGr < 5)

plot(data1to4$ppsqm ~ data1to4$BedroomAbvGr)
```




```
boxplot(ppsqm ~ BedroomAbvGr,data=data1to4)
```




```
require(nortest)
```

```
## Loading required package: nortest
```

```
bartlett.test(data1to4$ppsqm ~ data1to4$BedroomAbvGr)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  data1to4$ppsqm by data1to4$BedroomAbvGr
## Bartlett's K-squared = 17.446, df = 3, p-value = 0.0005722
```

```
lillie.test(data1to4$ppsqm)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  data1to4$ppsqm
## D = 0.017688, p-value = 0.3579
```

```
lillie.test(data1to4$ppsqm[data1to4$BedroomAbvGr == 1])
```

```
##
```

```
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  data1to4$ppsqm[data1to4$BedroomAbvGr == 1]
## D = 0.10279, p-value = 0.2586
```

```
lillie.test(data1to4$ppsqm[data1to4$BedroomAbvGr == 2])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  data1to4$ppsqm[data1to4$BedroomAbvGr == 2]
## D = 0.041909, p-value = 0.1456
```

```
lillie.test(data1to4$ppsqm[data1to4$BedroomAbvGr == 3])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  data1to4$ppsqm[data1to4$BedroomAbvGr == 3]
## D = 0.02798, p-value = 0.1365
```

```
lillie.test(data1to4$ppsqm[data1to4$BedroomAbvGr == 4])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  data1to4$ppsqm[data1to4$BedroomAbvGr == 4]
## D = 0.069636, p-value = 0.01426
```

```
aov = aov(data1to4$ppsqm ~ data1to4$BedroomAbvGr)

summary(aov)
```

```
##                          Df  Sum Sq Mean Sq F value Pr(>F)
## data1to4$BedroomAbvGr    1   89864   89864   121.7 <2e-16 ***
## Residuals             1399 1033254     739
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```